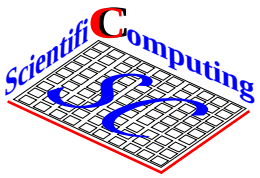
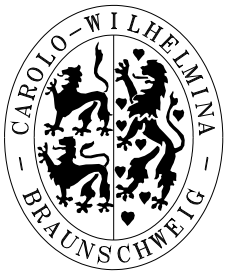


Nonlinear Galerkin Methods for the Model Reduction of Nonlinear Dynamical Systems

Hermann G. Matthies, Marcus Meyer
Institute of Scientific Computing
Technical University Braunschweig
Brunswick, Germany

Informatikbericht Nr.: 2002-3

March 2002



Nonlinear Galerkin Methods for the Model Reduction of Nonlinear Dynamical Systems

Hermann G. Matthies, Marcus Meyer
Department of Computer Science
Technical University Braunschweig
Brunswick, Germany

Informatikbericht Nr.: 2002-3

March 2002

Contributed to a special Issue of *Computers and Structures*.

Revised and expanded version of a contribution to the EUROMECH Coloquium 427, ENS Cachan France, September 2001

Location
Institute of Scientific Computing
Technische Universität Braunschweig
Hans-Sommer-Strasse 65
D-38106 Braunschweig

Postal Address
Institut für Wissenschaftliches Rechnen
Technische Universtität Braunschweig
D-38092 Braunschweig
Germany

Contact
Phone: +49-(0)531-391-3000
Fax: +49-(0)531-391-3003
E–Mail: wire@tu-bs.de

Nonlinear Galerkin Methods for the Model Reduction of Nonlinear Dynamical Systems

Hermann G. Matthies Marcus Meyer
Institute of Scientific Computing
Technische Universität Braunschweig
Brunswick, Germany
March 2002

Abstract

Numerical simulations of large nonlinear dynamical systems, especially over long time intervalls, may be computationally very expensive. Model reduction methods have been used in this context for a long time, usually projecting the dynamical system onto a subspace of its phase space. Nonlinear Galerkin methods try to improve on this by projecting onto a submanifold which does not have to be flat. These methods are applied to the finite element model of a windturbine, where both the mechanical and the aerodynamical degrees of freedom can be considered for model reduction. For the internal forces (moments, section forces) the nonlinear Galerkin method gives a considerable increase in accuracy for very little computational cost.

Keywords: Nonlinear Galerkin Method, Postprocessed Galerkin Method, Projection-based Model Reduction

1 Introduction

The application we are aiming at is the fatigue investigation of the wind turbine. It has become standard practice in the wind turbine industry to base the fatigue assesment on Monte-Carlo simulations of the wind turbine model in turbulent wind. Due to the frequency content of the wind spectrum and the requirements from the Monte-Carlo simulation, this involves simulating the wind turbine for a long time, i.e. many revolutions of the turbine and many periods of the relevant eigen-frequencies of the wind turbine. Thus it is of paramount importance to have efficient procedures to do this. Model reduction methods thus come naturally, as they allow to reduce the number of degrees of freedom for the ensuing time integration. The horizontal axis wind turbine will be modeled by a system of inteconnected beam finite element systems and is hence originally given by a system of partial differential equations. Instead of applying the nonlinear Galerkin

method to this original system, we shall first perform a finite element discretisation — a *flat* Galerkin projection — and apply our procedure to the resulting finite dimensional system. Due to geometrical and material modeling reasons the finite element model is rather detailed and far too large for long-time simulation of the system.

What is termed nonlinear Galerkin methods here has been used already in its simplest form for some while under other names, as in synergetics in the general context of reducing the behaviour of a system to the "important" or "essential" part, cf. HAKEN [17] for an account, and is called the "slaving principle" in that context. The mathematical background has subsequently been developed in the context of approximating — mainly dissipative — partial differential equations, cf. TEMAM [41]. Seen in an abstract setting, it may be regarded as approximating the dynamics of a system on the phase manifold by some projection onto a submanifold. The well-known Galerkin methods do essentially the same, but the submanifold is restricted at being a subspace, i.e. a *flat* submanifold. The nonlinear Galerkin method tries to improve on this by not restricting the submanifold to an affine subspace. It is a natural extension on the *flat* Galerkin method, in that it starts also from an approximation with a subspace. Now the additional variables, which will allow a *non-flat* approximating manifold, have to be defined.

There are several different ways in how these additional variables can be defined, and the simplest ones, also used in this work, will be explained in section 2. In many cases it may be assumed that the dynamics of the system is essentially only on a submanifold of phase space. These submanifolds, which may be the *centre* manifold near an equilibrium, or an *inertial* manifold, capture the main and essential dynamics of the system. Even if we can not show à priori that such a manifold exists, it has turned out to be advantageous in numerical calculations to just assume that it does exist, and to try to approximate it with an *approximate inertilal* manifold (AIM), which will be shown in subsection 2.1.

In any case, having performed the model reduction, we would like to estimate the error. In section 3 we show how this can be done with dual methods, which is essentially computing the Green's or influence function of the problem with respect to some functional of interest of the whole solution. For the sake of simplicity, the basic idea is first demonstrated in subsection 3.1 by applying it to a linear system of equations, and then extended to nonlinear systems in subsection 3.2 by linearisation. As we have both reduced the model to smaller dimensions and are also using a finite difference scheme in time in the actual numerical calculation, there are two distinct components of the error, which are analysed in section 4. The particular application of the combined method of model reduction controlled by error estimation by dual methods to a horizontal axis wind turbine is given in section 5, and some representative numerical results are shown in section 6.

In total we present a method which is capable of substantially reducing the

numerical effort involved in long time simulation of dynamical systems, where the approximation error incurred through the model reduction may be estimated and thus controlled.

2 The Nonlinear and Postprocessed Galerkin Method

As remarked earlier, we shall assume that we are dealing with a finite dimensional nonlinear dynamical system in order to avoid — for the sake of simplicity of exposition — questions of convergence. Where the system in question is originally modeled with a system of partial differential equations in space and time, we assume that it has already been discretised by some means in the spatial dimension. In our case this spatial semi-discretisation has been perfomed with finite elements. We start with an abstract setting, a nonlinear dynamical system where we seperate the linear and the higher order nonlinear part:

$$\dot{\boldsymbol{x}} + \boldsymbol{g}(\boldsymbol{x}, t) = \dot{\boldsymbol{x}} + \boldsymbol{A}\boldsymbol{x} + \boldsymbol{h}(\boldsymbol{x}, t) = \boldsymbol{0}, \qquad \boldsymbol{x} \in \mathbb{R}^d \tag{1}$$

with $\boldsymbol{A}\boldsymbol{x}$ as linear and $\boldsymbol{h}(\boldsymbol{x}, t)$ as nonlinear part of the system $\boldsymbol{g}(\boldsymbol{x}, t)$.

For the solution $\boldsymbol{x}$ we make the ansatz

$$\boldsymbol{x} = \boldsymbol{Y}_m \boldsymbol{\xi} + \boldsymbol{Z}_m \boldsymbol{\eta}., \tag{2}$$

where the columns of the matrix $\boldsymbol{Y}_m = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m]$ span the m-dimensional $\boldsymbol{A}$-invariant subspace $\mathcal{Y}_m = \text{span}\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m\}$ formed by the first m eigenvectors

$$\boldsymbol{A}\boldsymbol{y}_j = \lambda_j \boldsymbol{y}_j, \quad j = 1, \ldots, m, \tag{3}$$

and the columns of the matrix $\boldsymbol{Z}_m$ span the complementary subspace $\mathcal{Z}_m = \text{span}\{\boldsymbol{y}_{m+1}, \ldots, \boldsymbol{y}_d\}$ of the remaining $d - m$ eigenvectors. Usually $m \ll d$. Denote by $\tilde{\boldsymbol{Y}}_m$ and $\tilde{\boldsymbol{Z}}_m$ the corresponding matrices formed by the dual basis — the eigenvectors of $\boldsymbol{A}^T$.

Substituting from Eq. (2) into Eq. (1) and projecting onto the spaces $\mathcal{Y}_m$ and $\mathcal{Z}_m$ by multiplication with $\tilde{\boldsymbol{Y}}_m^T$ and $\tilde{\boldsymbol{Z}}_m^T$ from the left, we obtain two systems of differential equations, coupled through the nonlinear term:

$$\dot{\boldsymbol{\xi}} + \tilde{\boldsymbol{Y}}_m^T \boldsymbol{A}\boldsymbol{Y}_m \boldsymbol{\xi} + \tilde{\boldsymbol{Y}}_m^T \boldsymbol{h}(\boldsymbol{Y}_m \boldsymbol{\xi} + \boldsymbol{Z}_m \boldsymbol{\eta}, t) = \boldsymbol{0}, \tag{4}$$

$$\dot{\boldsymbol{\eta}} + \tilde{\boldsymbol{Z}}_m^T \boldsymbol{A}\boldsymbol{Z}_m \boldsymbol{\eta} + \tilde{\boldsymbol{Z}}_m^T \boldsymbol{h}(\boldsymbol{Y}_m \boldsymbol{\xi} + \boldsymbol{Z}_m \boldsymbol{\eta}, t) = \boldsymbol{0}. \tag{5}$$

The normal *flat* Galerkin method — which also uses $\boldsymbol{Y}_m$ for the weighting instead of $\tilde{\boldsymbol{Y}}_m$ — sets $\boldsymbol{\eta} = \boldsymbol{0}$ in Eq. (4) and integrates the resulting low-dimensional

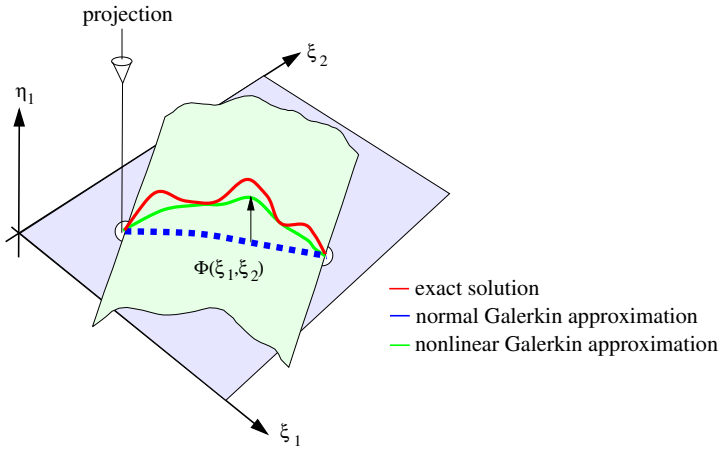


Figure 1: Geometric Interpretation of normal and nonlinear Galerkin Methods.

system

$$\dot{\boldsymbol{\xi}} + \tilde{\boldsymbol{Y}}_m^T \boldsymbol{A} \boldsymbol{Y} \boldsymbol{\xi} + \tilde{\boldsymbol{Y}}_m^T \boldsymbol{h}(\boldsymbol{Y}_m \boldsymbol{\xi}, t) = \boldsymbol{0}. \tag{6}$$

in time. The true solution of Eq. (1) is then approximated by $\boldsymbol{x} \approx \boldsymbol{Y}_m \boldsymbol{\xi}$.

Under the assumption that Eq. (1) has a low dimensional attractor, which can be described by a small number of degrees of freedom, we may assume that for a certain value of m and the correct separation of eigenmodes in $\boldsymbol{Y}_m$ and $\boldsymbol{Z}_m$ there exists a relation between $\boldsymbol{\xi}$ and $\boldsymbol{\eta}$, so that on the attractor the relation

$$\boldsymbol{\eta} = \boldsymbol{\Phi}(\boldsymbol{\xi}) \tag{7}$$

holds. If this is the case, the resulting manifold is an inertial manifold of the system Eq. (1), sketched in Fig. 1. This means that on the inertial manifold the behaviour of all the higher eigenmodes is governed by the behaviour of the m lower eigenmodes. In synergetics, cf. HAKEN [17], this is called the "slaving principle". By substituting Eq. (7) into Eq. (4) and integration of the resulting m-dimensional system

$$\dot{\boldsymbol{\xi}} + \tilde{\boldsymbol{Y}}_m^T \boldsymbol{A} \boldsymbol{Y}_m \boldsymbol{\xi} + \tilde{\boldsymbol{Y}}_m^T \boldsymbol{h}(\boldsymbol{Y}_m \boldsymbol{\xi} + \boldsymbol{Z}_m \boldsymbol{\Phi}(\boldsymbol{\xi}), t) = \boldsymbol{0}, \tag{8}$$

we obtain the reconstruction of the solution $\boldsymbol{x} = \boldsymbol{Y}_m \boldsymbol{\xi} + \boldsymbol{Z}_m \boldsymbol{\Phi}(\boldsymbol{\xi})$ *without* approximation error.

Many systems of differential equations are too complicated to prove the existence of an inertial manifold. In these cases the existence of an *approximate* inertial manifold (AIM) given by

$$\boldsymbol{\eta} \approx \boldsymbol{\Phi}_{app}(\boldsymbol{\xi}) \tag{9}$$

is postulated. The utility of this assumption is justified à posteriori by the error estimates and the reduced numerical work. Using the AIM and integrating

$$\dot{\boldsymbol{\xi}} + \tilde{\boldsymbol{Y}}_m^T \boldsymbol{A} \boldsymbol{Y}_m \boldsymbol{\xi} + \tilde{\boldsymbol{Y}}_m^T \boldsymbol{h}(\boldsymbol{Y}_m \boldsymbol{\xi} + \boldsymbol{Z}_m \boldsymbol{\Phi}_{app}(\boldsymbol{\xi}), t) = \boldsymbol{0} \tag{10}$$

in time, the solution is approximated by $\boldsymbol{x} \approx \boldsymbol{Y}_m \boldsymbol{\xi} + \boldsymbol{Z}_m \boldsymbol{\Phi}_{app}(\boldsymbol{\xi})$. This approximation will not exactly represent the dynamics on the attractor, but it is assumed that this solution is better than the solution of the standard Galerkin method.

A drawback of the original nonlinear Galerkin method is the effort to evaluate $\boldsymbol{\Phi}_{app}$ in Eq. (10), which can be costly. This raised the question if instead of solving equation Eq. (10) one could equally well use the traditional Galerkin method with $\hat{m} > m$ eigenmodes. This was addressed in the work of GARCÍA-ARCHILLA et al. [14], who developed a variant of the nonlinear Galerkin method, called the *postprocessed* Galerkin method: The authors show that it is often possible to obtain the accuracy of the system given by Eq. (10) with numerical effort corresponding to the integration of an m-dimensional Galerkin projection.

The idea is to split the eigenmodes into three groups, $\boldsymbol{Y}_m$ as before, but $\boldsymbol{Z}_m = [\boldsymbol{y}_{m+1}, \ldots, \boldsymbol{y}_{\hat{m}}]$ and $\hat{\boldsymbol{Z}}_m = [\boldsymbol{y}_{\hat{m}+1}, \ldots, \boldsymbol{y}_d]$. The exact solution could be described by

$$\boldsymbol{x} = \boldsymbol{Y}_m \boldsymbol{\xi} + \boldsymbol{Z}_m \boldsymbol{\eta} + \hat{\boldsymbol{Z}}_m \boldsymbol{\zeta}. \tag{11}$$

The variables associated with $\hat{\boldsymbol{Z}}_m$ are completely neglected by setting $\boldsymbol{\zeta} = \boldsymbol{0}$, and the procedure is applied as before with the appropriate redefinition of $\boldsymbol{Z}_m$. The system given by Eq. (6) is integrated, and the approximate inertial manifold is only used when output is needed, i.e. to "lift" the solution onto the AIM at the time instant t_n needed for output,

$$\boldsymbol{x}(t_n) \approx \boldsymbol{Y}_m \boldsymbol{\xi}(t_n) + \boldsymbol{Z}_m \boldsymbol{\Phi}_{app}(\boldsymbol{\xi}(t_n)). \tag{12}$$

This approach has been successfully applied to a problem in structural dynamics: LAING et al. [26] show for a vibrating shell that the postprocessed Galerkin method yields a similar accuracy as the nonlinear Galerkin method, but with less numerical effort. Linear variants of this procedure have long been known in structural dynamics under the term *static correction*, e.g. cf. [19].

2.1 Calculation of Approximate Inertial Manifolds

In any case, either for the full or the postprocessed nonlinear Galerkin method, we have to address the question of how to obtain the approximate inertial manifold $\boldsymbol{\Phi}_{app}$. A comprehensive summary regarding this question can be found in the paper by RUSSEL et al. [37]. Thus we limit ourselves to a short overview: Starting

from Eq. (5) one tries to find an approximation of the relation $\boldsymbol{\eta} = \boldsymbol{\Phi}_{app}(\boldsymbol{\xi})$. This means the equation

$$\dot{\boldsymbol{\eta}} + \tilde{\boldsymbol{Z}}_m^T \boldsymbol{A} \boldsymbol{Z}_m \boldsymbol{\eta} + \tilde{\boldsymbol{Z}}_m^T \boldsymbol{h}(\boldsymbol{Y}_m \boldsymbol{\xi} + \boldsymbol{Z}_m \boldsymbol{\eta}, t) = \boldsymbol{0} \tag{13}$$

has to be solved, at least approximately.

This could for example be done with simple methods of low order in time: The so-called Euler-Galerkin AIM [13] uses the implicit Euler method to discretise Eq. (13) in time. With the time step τ, the initial value $\boldsymbol{\eta} = \boldsymbol{0}$ and one fix-point iteration to find the solution of the resulting nonlinear system of equations we get the desired relation

$$\boldsymbol{\eta} = \boldsymbol{\Phi}_{app}(\boldsymbol{\xi}) = -\tau (\boldsymbol{I} + \tau \tilde{\boldsymbol{Z}}_m^T \boldsymbol{A} \boldsymbol{Z}_m)^{-1} \tilde{\boldsymbol{Z}}_m^T \boldsymbol{h}(\boldsymbol{Y}_m \boldsymbol{\xi}, t). \tag{14}$$

Another approach can be found in the work of TITI et al. [42]. The authors assume that the time derivative $\dot{\boldsymbol{\eta}}$ can be neglected (but not the variable $\boldsymbol{\eta}$ itself) and thus calculate a quasi-stationary AIM, the variables $\boldsymbol{\eta}$ follow the variables $\boldsymbol{\xi}$ instantaneously. In this case the algebraic equation

$$\boldsymbol{g}(\boldsymbol{\eta}) := \tilde{\boldsymbol{Z}}_m^T \boldsymbol{A} \boldsymbol{Z}_m \boldsymbol{\eta} + \tilde{\boldsymbol{Z}}_m^T \boldsymbol{h}(\boldsymbol{Y}_m \boldsymbol{\xi} + \boldsymbol{Z}_m \boldsymbol{\eta}, t) = \boldsymbol{0} \tag{15}$$

has to be solved. This means the system of differential equations (4) and (5) is replaced by a system of differential-algebraic equations (4) and (15). The authors further propose to use a fix-point iteration with one or two iterations to solve the algebraic equation. Alternatively one can use the Newton-Raphson iteration. Since the numerical effort to calculate $\boldsymbol{\Phi}_{app}$ should be as low as possible, one fix-point iteration is preferred [26]. In our calculations we use this last proposal, but with the redefinition of $\boldsymbol{Z}_m$ already mentioned.

3 Error Estimation Using Dual Methods

As the methods described in section 2 involve some element of approximation in order to allow the reduction of the numerical effort, it is desirable to be able to control the error incurred through this procedure. The main feature of the approach presented here is the use of the adjoint or dual problem. When combined with the methods of section 2 it allows the trade-off between accuracy and numerical effort to be estimated and thus adjusted to the actual needs.

Solution of the dual or adjoint problem, which means the solution of a linearised problem in order to compute a Green's or influence function, has long been used for the estimation of parameter sensitivity, e.g. in the process of design

optimisation [3, 20, 24]. But it has also been used in the estimation of the discretisation error of differential equations for use in adaptive methods in space and time.

Starting from the work of JOHNSON [21] and ERIKSSON et al. [10], where this approach is sometimes termed as *goal-oriented error estimation*, as it requires the user to give a functional of the solution — the goal — which is to be approximated, RANNACHER et al. [35, 5, 4, 1, 34, 6] have developed the so-called *dual-weighted-residual method* and used it for adaptive grid refinement in the finite-element discretisation of partial differential equations. This approach is also used in the error control of time stepping schemes for ordinary differential equations [11, 28, 27]. Additionally, the dual solution may be used to improve the already computed value of the desired functional [39, 15, 32]. For additional information concerning dual methods cf. KLEIBER [24] and MARCHUK [29].

3.1 Approximation of a Linear Problem

In order to introduce the method in its simplest setting, we shall — following [34] — apply it to a simple linear algebraic system. Assume that we have to solve

$$\boldsymbol{M}\boldsymbol{u} = \boldsymbol{b}, \quad \boldsymbol{u}, \boldsymbol{b} \in \mathbb{R}^d. \tag{16}$$

Let $\hat{\boldsymbol{M}}$ and $\hat{\boldsymbol{b}}$ be approximations of $\boldsymbol{M}$ and $\boldsymbol{b}$ of the system Eq. (16), such that $\hat{\boldsymbol{u}}$ satisfies the approximate equation:

$$\hat{\boldsymbol{M}}\hat{\boldsymbol{u}} = \hat{\boldsymbol{b}} \tag{17}$$

Also assume that the quantity of interest is a functional $J(\boldsymbol{u})$ of the solution. As we compute $J(\hat{\boldsymbol{u}})$, there is an error in the value of the functional compared with $J(\boldsymbol{u})$. In the simplest case J is linear and can thus be written as $J(\boldsymbol{u}) = \boldsymbol{p}^T\boldsymbol{u}$. Here the vector $\boldsymbol{p} \in \mathbb{R}^d$ defines the functional J, which has the approximation error

$$J(\boldsymbol{u}) - J(\hat{\boldsymbol{u}}) = J(\boldsymbol{e}) = \boldsymbol{p}^T\boldsymbol{e}, \tag{18}$$

where $\boldsymbol{e} = \boldsymbol{u} - \hat{\boldsymbol{u}}$ is the error in the solution and satisfies

$$\boldsymbol{M}\boldsymbol{e} = \boldsymbol{M}(\boldsymbol{u} - \hat{\boldsymbol{u}}) = \boldsymbol{M}\boldsymbol{u} - \boldsymbol{M}\hat{\boldsymbol{u}} = \boldsymbol{b} - \boldsymbol{M}\hat{\boldsymbol{u}} =: \boldsymbol{\rho}, \tag{19}$$

with the residuum $\boldsymbol{\rho}$. In order to estimate this error one may use the solution of the adjoint or dual problem: $\boldsymbol{M}^T\boldsymbol{\lambda} = \boldsymbol{p}$. With this we obtain:

$$J(\boldsymbol{e}) = \boldsymbol{p}^T\boldsymbol{e} = (\boldsymbol{M}^T\boldsymbol{\lambda})^T\boldsymbol{e} = \boldsymbol{\lambda}^T\boldsymbol{M}\boldsymbol{e} = \boldsymbol{\lambda}^T\boldsymbol{\rho}. \tag{20}$$

Hence we obtain an *à posteriori* error estimate

$$|J(\boldsymbol{e})| \leq \sum_{i=1}^{d} |\rho_i| \, |\lambda_i| \tag{21}$$

with the "local" residuum ρ_i, weighted with the "local" dual solution λ_i.

3.2 Approximation of a Nonlinear Problem

Extension to nonlinear systems and nonlinear functionals is immediate via linearisation: Assume that we want to solve $\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{0}$, where $\boldsymbol{F}$ is some nonlinear function; and assume additionally that instead we solve some approximation $\hat{\boldsymbol{F}}(\hat{\boldsymbol{u}}) = \boldsymbol{0}$. The functional of interest is still denoted by $J(\boldsymbol{u})$, now nonlinear with derivative $DJ(\boldsymbol{u})$. The dual or adjoint problem is again linear:

$$(D\boldsymbol{F}(\hat{\boldsymbol{u}}))^T \boldsymbol{\lambda} = DJ(\hat{\boldsymbol{u}})^T, \tag{22}$$

where $D\boldsymbol{F}(\hat{\boldsymbol{u}})$ is the derivative (Jacobian matrix) of $\boldsymbol{F}$. The error in the functional may be estimated by linearisation $J(\boldsymbol{u}) - J(\hat{\boldsymbol{u}}) \approx DJ(\hat{\boldsymbol{u}})(\boldsymbol{u} - \hat{\boldsymbol{u}})$, and from Eq. (22) we obtain

$$J(\boldsymbol{u}) - J(\hat{\boldsymbol{u}}) \approx ((D\boldsymbol{F}(\hat{\boldsymbol{u}}))^T \boldsymbol{\lambda})^T (\boldsymbol{u} - \hat{\boldsymbol{u}}) = \boldsymbol{\lambda}^T D\boldsymbol{F}(\hat{\boldsymbol{u}})(\boldsymbol{u} - \hat{\boldsymbol{u}}) \tag{23}$$

Now, again by linearisation $D\boldsymbol{F}(\hat{\boldsymbol{u}})(\boldsymbol{u} - \hat{\boldsymbol{u}}) \approx \boldsymbol{F}(\boldsymbol{u}) - \boldsymbol{F}(\hat{\boldsymbol{u}})$; and as $\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{0}$, we have from Eq. (23)

$$J(\boldsymbol{u}) - J(\hat{\boldsymbol{u}}) \approx -\boldsymbol{\lambda}^T \boldsymbol{F}(\hat{\boldsymbol{u}}) = \boldsymbol{\lambda}^T \boldsymbol{\rho}, \tag{24}$$

with the residuum $\boldsymbol{\rho} := -\boldsymbol{F}(\hat{\boldsymbol{u}})$. Hence we obtain at least approximately the *à posteriori* error estimate of Eq. (21).

4 Errors from Model-Reduction and Time-Discretisation

After these general considerations in the last section 3, we turn again to our problem at hand, namely to analyse the error incurred due to use of the nonlinear Galerkin method. For linear problems this has been considered frequently in the literature, e.g. the error due to projection onto a combined modal and Lanczos-basis is considered in KLINE [25]. He comes to the result that the error is composed of two components: One is caused from the projection of the external loading onto the reduced basis, and the other one is due to the restricted ability of the reduced basis to represent the dynamics of the system.

JOO, WILSON and LEGER [23] propose criteria for the dimension of the Lanczos-basis and also show, that loading components which are orthogonal to the basis are relevant for magnitude of the error. IBRAHIMBEGOVIĆ and WILSON [19] also consider the choice of basis depending on the spatial distribution of the loading as well as the frequency content, and give à priori criteria for the choice of basis, both for a modal basis and for Lanczos-vector basis. CABOS [7] gives à posteriori error bounds for linear vibration problems with harmonic excitation, which are solved with Krylov subspace techniques using a Lanczos basis. The error bound given for the error in a linear functional is very similar to the one considered here.

In the following an error estimate based on the dual methods from section 3 will be derived. It contains both the error due to use of the nonlinear Galerkin method (model reduction), and the one due to time-discretisation.

Here we use the $cG(1)$ method (continuous Galerkin with peace-wise linear time functions) for time-discretisation, which may under certain conditions be shown to be equivalent to the NEWMARK method [31]. In our analysis, we follow JOHNSON [22]. Starting again with Eq. (1)

$$\begin{aligned} &\dot{\boldsymbol{x}} + \boldsymbol{g}(\boldsymbol{x}, t) = \dot{\boldsymbol{x}} + \boldsymbol{A}x + \boldsymbol{h}(\boldsymbol{x}, t) = \boldsymbol{0}, \quad 0 < t \leq T, \\ &\boldsymbol{x}(0) = \boldsymbol{x}_0, \quad \boldsymbol{x} \in \mathbb{R}^d, \end{aligned} \tag{25}$$

we perform the reduction $\boldsymbol{x} \approx \boldsymbol{x}_m = \boldsymbol{Y}_m \boldsymbol{\xi}$ to arrive at

$$\begin{aligned} &\dot{\boldsymbol{\xi}} + \tilde{\boldsymbol{Y}}_m^T \boldsymbol{g}(\boldsymbol{Y}_m \boldsymbol{\xi}, t) = \dot{\boldsymbol{\xi}} + \tilde{\boldsymbol{g}}_m(\boldsymbol{\xi}, t) = \boldsymbol{0}, \\ &\boldsymbol{\xi}(0) = \boldsymbol{Y}_m^T \boldsymbol{x}(0). \end{aligned} \tag{26}$$

To perform the $cG(1)$ discretisation, we look for an approximation $\boldsymbol{\xi}_k$ of $\boldsymbol{\xi}$, which is continuous and piece-wise linear, and satisfies

$$\int_{I_n} \left[\dot{\boldsymbol{\xi}}_k + \boldsymbol{g}_m(\boldsymbol{\xi}_k, t) \right] \, dt = 0 \tag{27}$$

for each time interval I_n, $n = 1, \ldots, N$. In order to write this equation completely in discrete form, a quadrature method for the integral has to be chosen. Here we select the trapezoidal rule and obtain for each time interval the equation

$$\boldsymbol{\xi}_{k,n} - \boldsymbol{\xi}_{k,n-1} + \frac{\Delta t_n}{2} \left[\boldsymbol{g}_m(\boldsymbol{\xi}_{k,n-1}, t_{n-1}) + \boldsymbol{g}_m(\boldsymbol{\xi}_{k,n}, t_n) \right] = 0, \tag{28}$$

with $\boldsymbol{\xi}_{k,0} = \boldsymbol{\xi}(0)$. With $\boldsymbol{x}_{m,k} = \boldsymbol{Y}_m \boldsymbol{\xi}_k$ the residuum is given by the expression

$$\boldsymbol{\rho}(t) = - \left[\boldsymbol{Y}_m \dot{\boldsymbol{\xi}}_k + \boldsymbol{g}(\boldsymbol{Y}_m \boldsymbol{\xi}_k, t) \right]. \tag{29}$$

Let $\boldsymbol{\lambda}$ be the solution of the adjoint equation (backward in time)

$$\begin{aligned} -\dot{\boldsymbol{\lambda}} + [\boldsymbol{A}(t)]^T \boldsymbol{\lambda} &= \boldsymbol{p}(t), \quad T > t \geq 0 \\ \boldsymbol{\lambda}(T) &= \mathbf{0}, \end{aligned} \tag{30}$$

where $\boldsymbol{p}(t)$ defines the linearisation of some functional of the form $J(\boldsymbol{x}) = \int_0^T j(\boldsymbol{x}(t), t)\, dt$, its action on $\boldsymbol{q}(t)$ given by

$$DJ(\boldsymbol{x})\boldsymbol{q} = \langle \boldsymbol{p}, \boldsymbol{q} \rangle = \int_0^T \boldsymbol{p}(t)^T \boldsymbol{q}(t)\, dt. \tag{31}$$

The error in the functional may now be written as

$$\begin{aligned} J(\boldsymbol{x}) - J(\boldsymbol{x}_{m,k}) &\approx \int_0^T \boldsymbol{p}^T (\boldsymbol{x} - \boldsymbol{x}_{m,k})\, dt \\ &= \int_0^T -(\dot{\boldsymbol{\lambda}} + [\boldsymbol{A}(t)]^T \boldsymbol{\lambda})^T (\boldsymbol{x} - \boldsymbol{x}_{m,k})\, dt \\ &\approx \int_0^T \boldsymbol{\rho}(t)^T \boldsymbol{\lambda}(t)\, dt + \boldsymbol{\lambda}(0)^T (\boldsymbol{x}(0) - \boldsymbol{x}_{m,k}(0)) \\ &= \sum_{n=1}^N \int_{I_n} \boldsymbol{\rho}^T \boldsymbol{\lambda}\, dt + \boldsymbol{\lambda}(0)^T (\boldsymbol{x}(0) - \boldsymbol{x}_{m,k}(0)) \\ &= \sum_{n=1}^N \sum_{j=1}^d \int_{I_n} \rho_j \lambda_j\, dt + \boldsymbol{\lambda}(0)^T (\boldsymbol{x}(0) - \boldsymbol{x}_{m,k}(0)) \\ &\leq \sum_{n=1}^N \sum_{j=1}^d \sup_{t \in I_n} |\rho_j| \int_{I_n} \lambda_j\, dt + \boldsymbol{\lambda}(0)^T (\boldsymbol{x}(0) - \boldsymbol{x}_{m,k}(0)). \end{aligned}$$

The parts due to model-reduction and time-discretisation as well as those due to the approximation of the initial value may be clearly distinguished.

The solution $\boldsymbol{\lambda}$ of the dual problem, which can also only be done approximately, has to use smaller time steps, or higher interpolation order in the cG time stepping procedure, or both.

5 Application to a Nonlinear Dynamical System

As application we consider the problem of simulating the dynamic behaviour of a wind turbine in turbulent wind. The wind turbine consists of a tower, including the nacelle, and three blades. Both tower and blades are discretised spatially using the finite element method. In Fig. 2 a visualisation of the problem and in Fig. 3

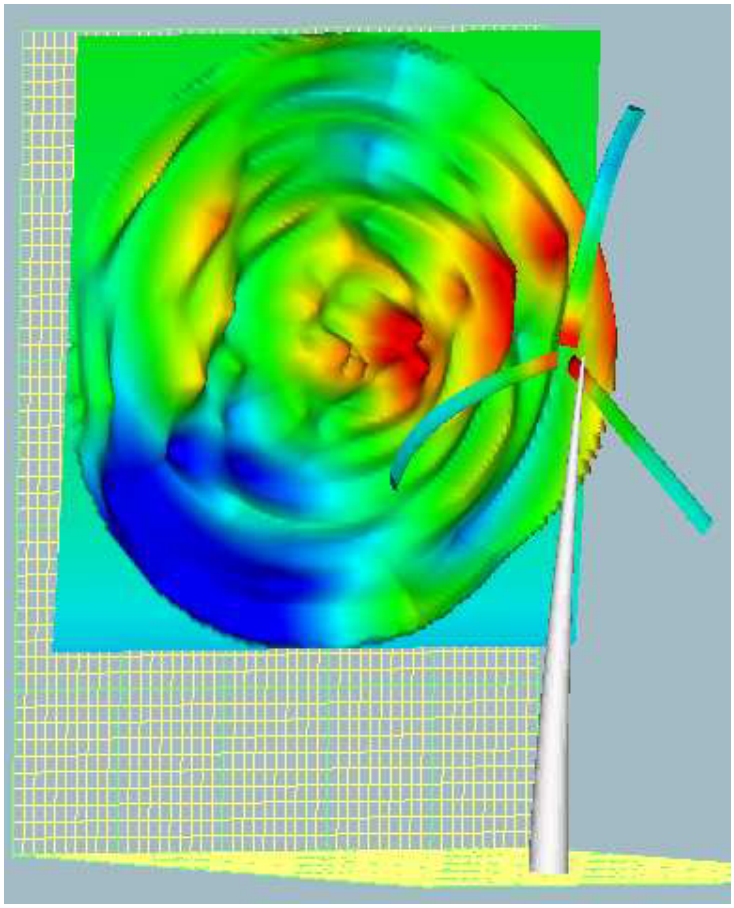


Figure 2: Wind turbine in turbulent wind

the structural model is shown. To account for large deformations, non-linear beam elements are used to discretise both tower and blades. In the literature a number of non-linear three-dimensional beam elements can be found, see e.g. [38], [8], [18], [33], [9], mostly differing in the parameterisation and interpolation of rotations in three-dimensional space. In our case the implementation follows [8] and [16]; the elements have two nodes with six degrees of freedom $\boldsymbol{d} = (\boldsymbol{u}, \boldsymbol{\theta})^T$ each, where $\boldsymbol{u} = (u_x, u_y, u_z)^T$ describes the translation in the inertial reference frame and $\boldsymbol{\theta} = (\theta_x, \theta_y, \theta_z)^T$ are the three components of the rotation vector. The inertia force, given by

$$\boldsymbol{f}_{kin} = \boldsymbol{M}(\boldsymbol{d})\ddot{\boldsymbol{d}} + \hat{\boldsymbol{f}}_{kin}(\dot{\boldsymbol{d}}, \boldsymbol{d}), \tag{32}$$

is expressed in terms of the first and second time derivative of the nodal variables, denoted by $\dot{\boldsymbol{d}}$ and $\ddot{\boldsymbol{d}}$ respectively. $\boldsymbol{M}$ denotes the mass-inertia matrix and $\hat{\boldsymbol{f}}_{kin}$

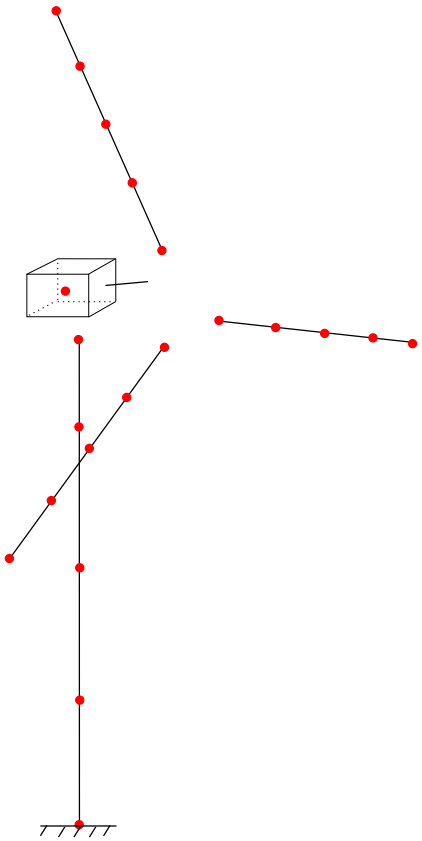


Figure 3: Schematic view of the structural model.

is a non-linear term describing the Coriolis and centrifugal forces. The nonlinear internal force of the beam element is denoted by $\boldsymbol{f}_{in}(\boldsymbol{d})$. A consistent linearisation of both internal and inertial force is obtained using a computer algebra program. The nonlinear external forces, i.e. aerodynamic and gravity forces, are denoted by $\boldsymbol{f}_{ext}(\ddot{\boldsymbol{d}}, \dot{\boldsymbol{d}}, \boldsymbol{d}, t)$. For the sake of brevity, and as it is not central to the subject at hand, we shall not address the complex nature of the aerodynamic forces and the methods for their efficient calculation in this paper. A detailed account of the procedure will be given elsewhere. The interested reader is referred to SPERA [40] and the references therein.

We obtain the equations of motion

$$
\begin{aligned}
&\boldsymbol{f}(\ddot{\boldsymbol{d}}, \dot{\boldsymbol{d}}, \boldsymbol{d}, t) = \boldsymbol{0}, \\
&\boldsymbol{d}(0) = \boldsymbol{d}_0, \quad \dot{\boldsymbol{d}}(0) = \boldsymbol{v}_0, \\
&\boldsymbol{d} \in \mathbb{R}^d, \quad t \in [0, T],
\end{aligned}
\tag{33}
$$

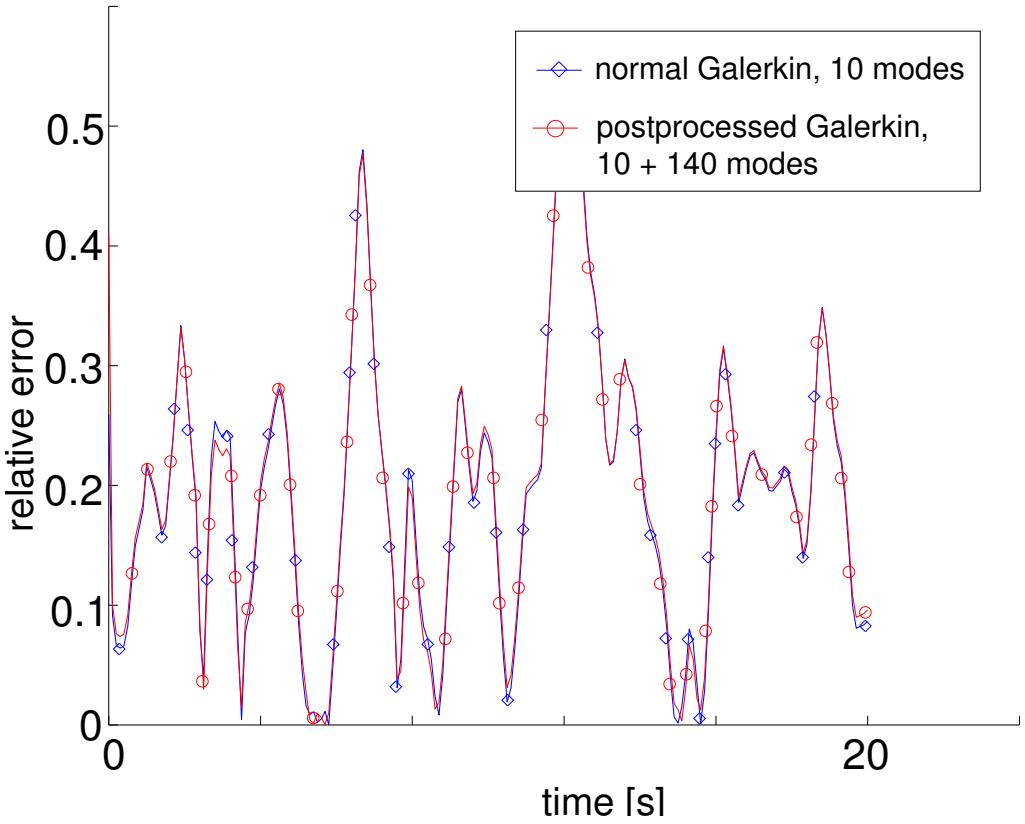


Figure 4: Relative displacement error, 10 primary and 140 secondary modes.

with

$$\boldsymbol{f} = \boldsymbol{f}_{kin}(\ddot{\boldsymbol{d}}, \dot{\boldsymbol{d}}, \boldsymbol{d}) + \boldsymbol{f}_{in}(\boldsymbol{d}) - \boldsymbol{f}_{ext}(\ddot{\boldsymbol{d}}, \dot{\boldsymbol{d}}, \boldsymbol{d}, t).$$

Since the aim is to perform long-term simulations under turbulent wind conditions, a reduction of the number of degrees of freedom is necessary to reduce the simulation time. We adopt an approach described e.g. in [2, 30, 16, 36] and references therein, the model reduction on substructure level. In the case of the wind turbine the substructures are the tower with nacelle and the three blades. For the following analysis we formulate the equations of motion of a substructure as

$$\boldsymbol{M}(\boldsymbol{d})\ddot{\boldsymbol{d}} + \boldsymbol{K}_0\boldsymbol{d} + \boldsymbol{h}(\ddot{\boldsymbol{d}}, \dot{\boldsymbol{d}}, \boldsymbol{d}, t) = \boldsymbol{0}, \tag{35}$$

with $\boldsymbol{K}_0 = D\boldsymbol{f}_{in}(\boldsymbol{d}_0)$ and $\boldsymbol{h} = \hat{\boldsymbol{f}}_{kin} + \boldsymbol{f}_{in} - \boldsymbol{K}_0\boldsymbol{d} - \boldsymbol{f}_{ext}$.

We calculate the modal basis of each substructure by solving the generalised eigenvalue problem

$$\boldsymbol{K}_0\boldsymbol{y} = -\omega^2\boldsymbol{M}(\boldsymbol{d}_0)\boldsymbol{y} \tag{36}$$

to obtain the eigenvectors $\boldsymbol{y}_j$ and eigenvalues ω_j^2. The modal basis is thus defined as

$$\boldsymbol{Y}_m = \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m\}, \quad m \ll d. \tag{37}$$

Substituting the reduced ansatz

$$\boldsymbol{d} \approx \boldsymbol{d}_m = \sum_{j=1}^{m} \boldsymbol{y}_j \xi_j = \boldsymbol{Y}_m \boldsymbol{\xi} \tag{38}$$

into the equations of motion and choosing the space spanned by the vectors $\boldsymbol{y}_j$ as test space, the reduced system of dimension m is given by

$$\boldsymbol{Y}_m^T \boldsymbol{f}(\boldsymbol{Y}_m \ddot{\boldsymbol{\xi}}, \boldsymbol{Y}_m \dot{\boldsymbol{\xi}}, \boldsymbol{Y}_m \boldsymbol{\xi}, t) = \boldsymbol{0}. \tag{39}$$

Employing the implicit Newmark-scheme for time integration [31] and the standard Newton-Raphson iteration the algorithm to calculate the solution of time step $n+1$ in the case of the normal Galerkin Method is given by

Set $i = 0$, calculate initial guess $\boldsymbol{d}_{n+1}^{(i)}, \boldsymbol{v}_{n+1}^{(i)}, \boldsymbol{a}_{n+1}^{(i)}$
Calculate $\boldsymbol{\rho}_{\text{red}}^{(i)}$ and $\boldsymbol{K}_{\text{eff,red}}^{(i)}$
  Start iteration, $i = i + 1$
  Solve $\Delta \boldsymbol{d}^{(i)} = -\boldsymbol{Y}_m^T (\boldsymbol{K}_{\text{eff,red}}^{(i-1)})^{-1} \boldsymbol{\rho}_{\text{red}}^{(i-1)}$
  Update Variables $\boldsymbol{d}_{n+1}^{(i)}, \boldsymbol{v}_{n+1}^{(i)}, \boldsymbol{a}_{n+1}^{(i)}$
  Calculate $\boldsymbol{\rho}_{\text{red}}^{(i)}$ and $\boldsymbol{K}_{\text{eff,red}}^{(i)}$
  If $\|\boldsymbol{\rho}_{\text{red}}^{(i)}\| \leq \epsilon \|\boldsymbol{\rho}_{\text{red}}^{(0)}\| \rightarrow$ stop iteration
Start calculation for next time step.

Here $\boldsymbol{K}_{\text{eff,red}}$ denotes the reduced tangent matrix given as

$$\boldsymbol{K}_{\text{eff,red}} = \boldsymbol{Y}_m^T \left[\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{a}_{n+1}} \frac{\partial \boldsymbol{a}_{n+1}}{\partial \boldsymbol{d}_{n+1}} + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{v}_{n+1}} \frac{\partial \boldsymbol{v}_{n+1}}{\partial \boldsymbol{d}_{n+1}} + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{d}_{n+1}} \right] \boldsymbol{Y}_m, \tag{40}$$

$\boldsymbol{\rho}_{\text{red}}^{(i)}$ the reduced residual

$$\boldsymbol{\rho}_{\text{red}}^{(i)} = \boldsymbol{Y}_m^T \left[\boldsymbol{f}(\boldsymbol{a}_{n+1}^{(i)}, \boldsymbol{v}_{n+1}^{(i)}, \boldsymbol{d}_{n+1}^{(i)}, t_{n+1}) \right], \tag{41}$$

and $\boldsymbol{a}_{n+1}^{(i)}$, $\boldsymbol{v}_{n+1}^{(i)}$ and $\boldsymbol{d}_{n+1}^{(i)}$ the approximations of $\ddot{\boldsymbol{d}}$, $\dot{\boldsymbol{d}}$ and $\boldsymbol{d}$ at time instant t_{n+1} respectively.

In the case of the postprocessed Galerkin method together with the quasi-static AIM the algorithm changes to

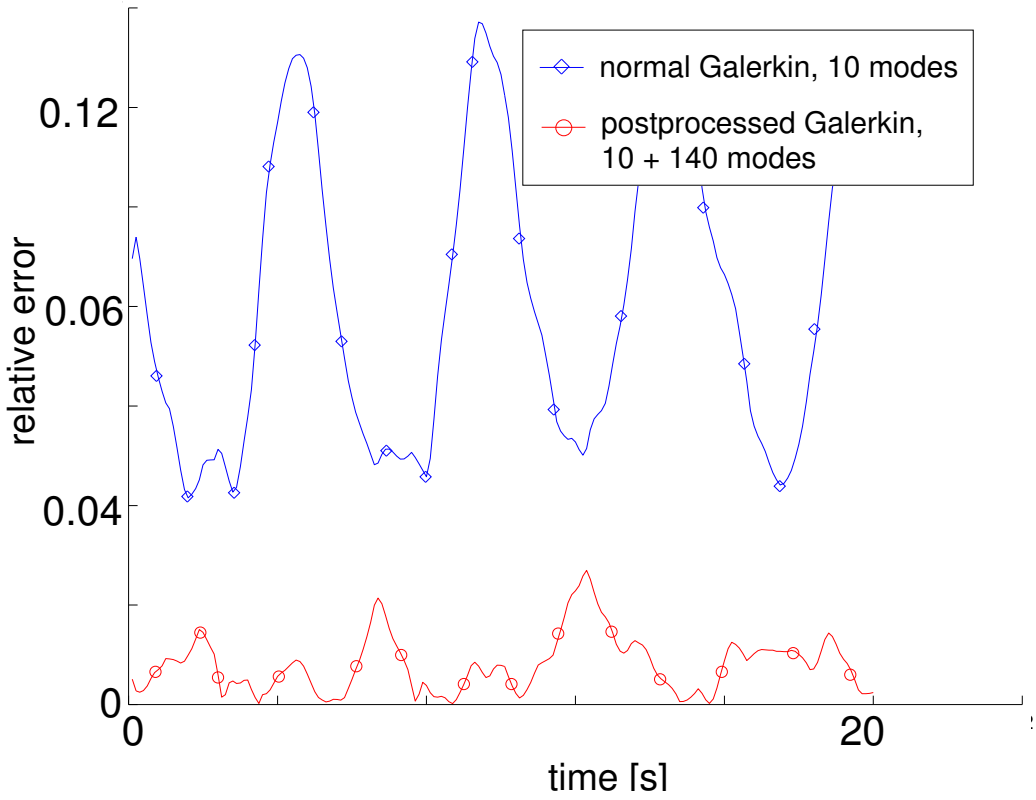


Figure 5: Relative bending moment error, 10 primary and 140 secondary modes

Set $i = 0$, calculate initial guess $\boldsymbol{d}_{n+1}^{(i)}, \boldsymbol{v}_{n+1}^{(i)}, \boldsymbol{a}_{n+1}^{(i)}$
Calculate $\boldsymbol{\rho}_{\text{red}}^{(i)}$ and $\boldsymbol{K}_{\text{eff,red}}^{(i)}$
    Start iteration, $i = i + 1$
    Solve $\Delta \boldsymbol{d}^{(i)} = -\boldsymbol{Y}_m^T (\boldsymbol{K}_{\text{eff,red}}^{(i-1)})^{-1} \boldsymbol{\rho}_{\text{red}}^{(i-1)}$
    Update Variables $\boldsymbol{d}_{n+1}^{(i)}, \boldsymbol{v}_{n+1}^{(i)}, \boldsymbol{a}_{n+1}^{(i)}$
    Calculate $\boldsymbol{\rho}_{\text{red}}^{(i)}$ and $\boldsymbol{K}_{\text{eff,red}}^{(i)}$
    If $\|\boldsymbol{\rho}_{\text{red}}^{(i)}\| \leq \epsilon \|\boldsymbol{\rho}_{\text{red}}^{(0)}\| \rightarrow$ stop iteration
Solve AIM$(\boldsymbol{d}_{n+1}^{(i)}, \boldsymbol{v}_{n+1}^{(i)}, \boldsymbol{a}_{n+1}^{(i)})$
Calculate $\hat{\boldsymbol{d}}_{n+1}^{(i)} = \boldsymbol{d}_{n+1}^{(i)}$+AIM
Start calculation for next time step.

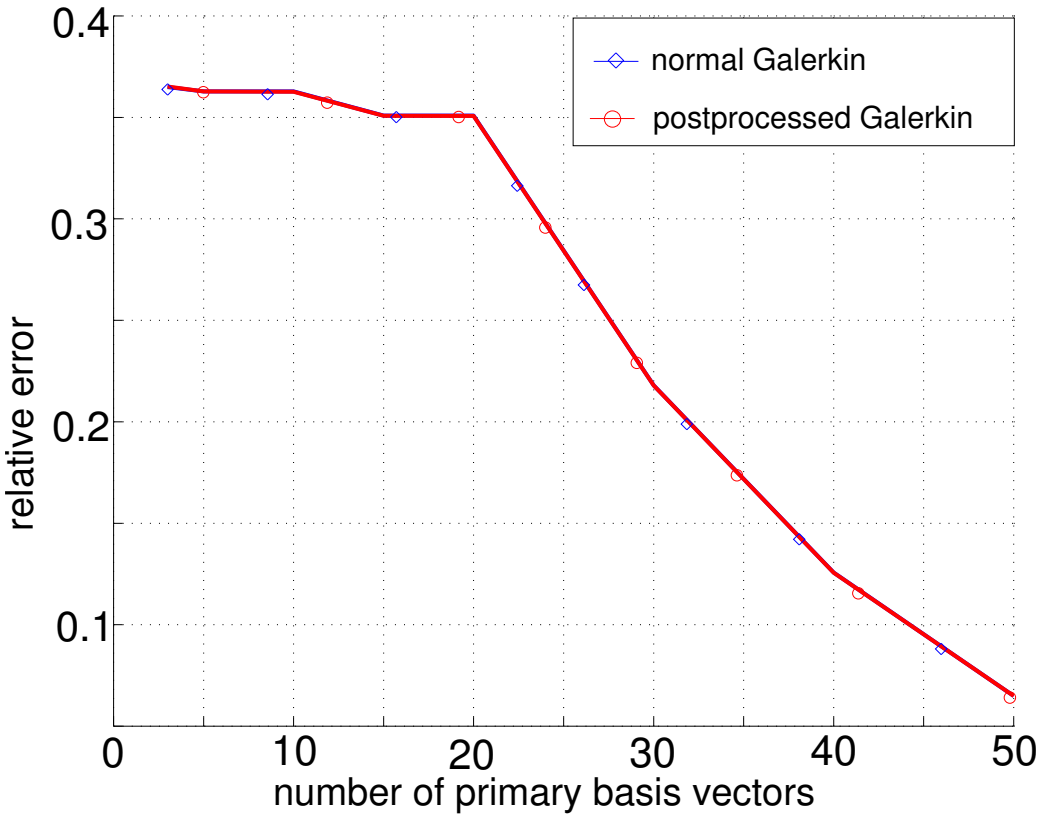


Figure 6: Mean values of relative displacement error for normal and postprocessed Galerkin Methods.

6 Numerical Results

Tower and blades of the wind turbine are discretised with the nonlinear beam elements mentioned above. In the following we show the results of simulations in turbulent wind, concentrating on one blade as a representative substructure. The lowest 150 eigenmodes of the blade are calculated and sorted according to eigenfrequency in ascending order. The reduction is performed using $m = [5, 10, 20, \ldots, 50]$ eigenmodes as primary variables and using the rest $150 - m$ eigenvectors as secondary variables in the postprocessing step. To analyse the approximation error of the reduced model we use the relative error of the displacements,

$$e_{u,m}(t) = \frac{\|\boldsymbol{u}_{\text{ref}} - \boldsymbol{u}_m\|}{\|\boldsymbol{u}_{\text{ref}}\|}, \tag{42}$$

and the relative error of the vector of bending moments,

$$e_{mb,m}(t) = \frac{\|\boldsymbol{mb}_{\text{ref}} - \boldsymbol{mb}_m\|}{\|\boldsymbol{mb}_{\text{ref}}\|}, \tag{43}$$

as a function of the dimension m of the basis.

The reference values $\boldsymbol{u}_{ref}$ and $\boldsymbol{mb}_{\text{ref}}$ are calculated using all degrees of freedom of the finite element model. In figures Fig. 4 and Fig. 5 a short time period of the calculated relative errors is shown, using $m = 10$ primary modes for the normal and 140 additional modes for the postprocessing step. Due to the turbulent wind the relative error is of stochastic nature and it is better to compare the mean value of the relative error: Calculating the mean value for different numbers of m and keeping the total number of 150 basis vectors, we obtain the results depicted in figure Fig. 6 for the displacement error and figure Fig. 7 for the bending moment error. We see that the postprocessing step does not lead to a significant decrease of the relative displacement error. But the error in the bending moments, which is particularly important for evaluating structural safety, is significantly reduced.

7 Summary

The nonlinear Galerkin method was employed on the structural model of a horizontal axis wind turbine in order to be able to do long time integration with as few degrees of freedom as possible. The error incurred through this procedure can be controlled by use of dual methods. Particularly the postprocessed version of the nonlinear Galerkin method needs only insignificantly more computation than the flat Galerkin scheme. Internal reaction forces were always more accurate with the nonlinear Galerkin method, while the error of the displacement sometimes was not changed significantly.

The failure of the postprocessed Galerkin method to increase the accuracy of the displacements is attributed to the smoothness of the forcing of the considered aero-elastic problem. This is in accordance with findings where the same problem of forced shell vibrations was investigated with smooth forcing (FOALE et al. [12]) and point-forcing (LAING et al. [26]). Only in the case of point-forcing a significant increase of accuracy of the displacements using the nonlinear Galerkin Methods was found. The authors conclude that in the case of smooth forcing and therefore smooth solutions the application of nonlinear Galerkin Methods is not necessary. In this paper we have shown that even in the case of smooth forcing the application of the postprocessed Galerkin Method can lead to a significant increase of the accuracy of derived quantities like bending moments. These results justify the extra cost of computing the postprocessing step.

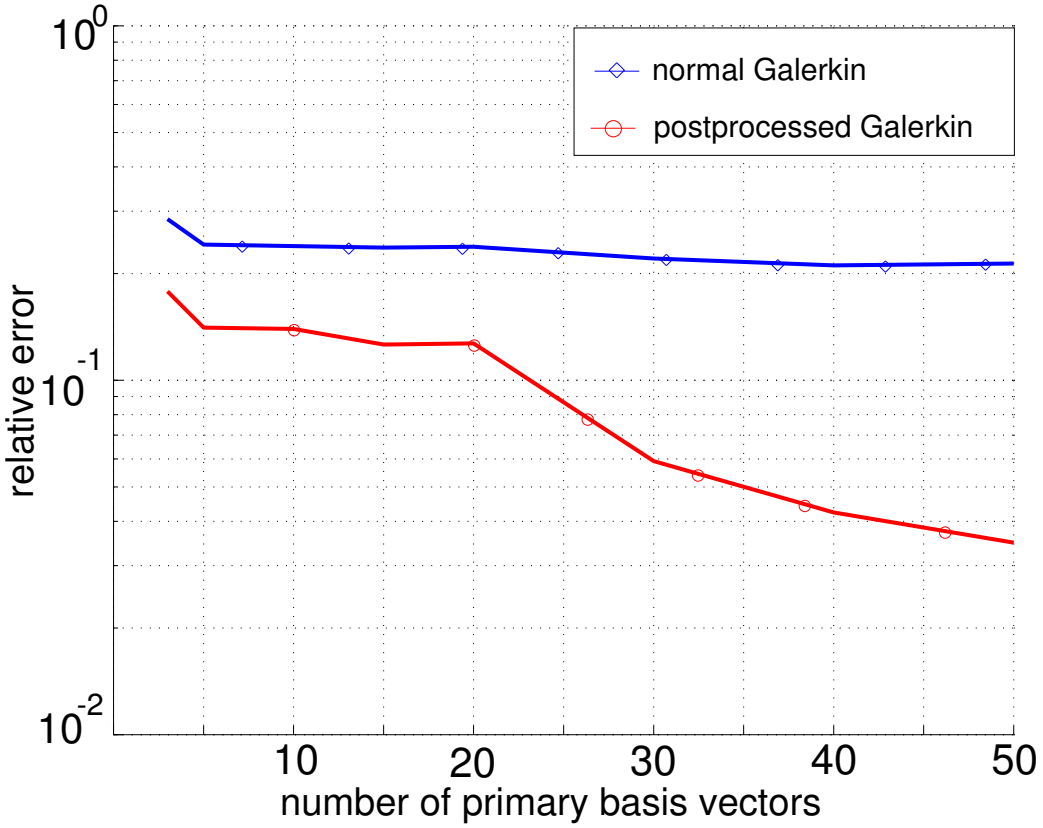


Figure 7: Mean values of relative bending moment error for normal and postprocessed Galerkin Methods.

References

[1] W. Bangert and R. Rannacher. Finite element approximation of the acoustic wave equation: Error control and mesh adaptation. *East-West Journal of Numerical Mathematics*, 7(4):263–282, 1999.

[2] K.-J. Bathe and S. Gracewski. On nonlinear dynamic analysis using substructuring and mode superposition. *Computers & Structures*, 13:699–707, 1981.

[3] O. Baysal and M.E. Eleshaky. Aerodynamic sensitivity analysis methods for the compressible Euler equations. *Journal of Fluids Engineering*, 113:681–688, 1991.

[4] R. Becker and R. Rannacher. A feed-back approach to error control in finite element methods: Basic analysis and examples. *East-West Journal for Numerical Mathematics*, 4:237–264, 1996.

[5] R. Becker and R. Rannacher. Weighted a posteriori error control in finite element methods. Technical report, Universität Heidelberg, 1996.

[6] R. Becker and R. Rannacher. An optimal control approach to a posteriori error estimation in finite elements. *Acta Numerica*, 10:1–102, 2001.

[7] C. Cabos. Error bounds for dynamic responses in forced vibration problems. *SIAM Journal of Scientific Computing*, 15(1):1–15, 1994.

[8] A. Cardona and M. Géradin. A beam finite element non-linear theory with finite rotations. *International Journal of Numerical Methods in Engineering*, 26:2403–2438, 1988.

[9] M.A. Crisfield and G. Jelenić. Objectivity of strain measures in the geometrically exact three-dimensional beam theory and its finite-element implementation. *Proceedings of the Royal Society London A*, 455:1125–1147, 1999.

[10] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. Introduction to adaptive methods for differential equations. *Acta Numerica*, 4:105–158, 1995.

[11] D. Estep. A posteriori error bounds and global error control for approximations of ordinary differential equations. *SIAM Journal of Numerical Analysis*, 32:1–48, 1995.

[12] S. Foale, J.M.T. Thompson, and F.A. McRobie. Numerical dimension-reduction methods for nonlinear shell vibrations. *Journal of Sound and Vibration*, 215(3):527–545, 1998.

[13] C. Foias, M.S. Jolly, I.G. Kevrekidis, G.R. Sell, and E.S. Titi. On the computation of inertial manifolds. *Physics Letters A*, 131(7/8):433–436, 1988.

[14] B. García-Archilla, J. Novo, and E.S. Titi. Postprocessing the Galerkin method: A novel approach to approximate inertial manifolds. *SIAM Journal of Numerical Analysis*, 35(3):941–972, 1998.

[15] M.B. Giles. On adjoint equations for error analysis and optimal grid adaptation in CFD. Technical report, Oxford University, 1997.

[16] M. Géradin et al. Finite element modeling concepts in multibody dynamics. In M.S. Pereira and J.A.C. Ambrosio, editors, *Computer-aided Analysis of Rigid and Flexible Mechanical Systems*, pages 233–284, Dordrecht, 1994. Kluwer Academic Publishers.

[17] H. Haken. *Advanced Synergetics*. Springer Verlag, Berlin, 1983.

[18] A. Ibrahimbegović and M. Al Mikad. Finite rotations in dynamics of beams and implicit time-stepping schemes. *International Journal for Numerical Methods in Engineering*, 41:781–814, 1998.

[19] A. Ibrahimbegović and E.L. Wilson. Automated truncation of Ritz vector basis in modal transformation. *ASCE Journal of Engineering Mechanics*, 116(11):2506–2520, 1990.

[20] A. Jameson. Aerodynamic design via control theory. *SIAM Journal of Scientific Computing*, 3:233–260, 1988.

[21] C. Johnson. Error estimates and adaptive time-step control for a class of one-step methods for stiff ordinary differential equations. *SIAM Journal of Numerical Analysis*, 25(4):908–926, 1988.

[22] C. Johnson. Adaptive computational methods for differential equations. In J.M. Ball and J.C.R. Hunt, editors, *ICIAM*, Oxford, 1999.

[23] K.J. Joo, E.L. Wilson, and P. Leger. Ritz vectors and generation criteria for mode superposition analysis. *Earthquake Engineering and Structural Dynamics*, 18:149–167, 1989.

[24] M. Kleiber. *Parameter Sensitivity in Nonlinear Mechanics*. John Wiley & Sons, Chichester, 1997.

[25] K.A. Kline. Dynamic analysis using a reduced basis of exact modes and Ritz vectors. *AIAA Journal*, 24(12):2022–2029, 1986.

[26] C.R. Laing, A. McRobie, and J.M.T. Thompson. The post-processed galerkin method applied to non-linear shell vibrations. *Dynamics and Stability of Systems*, 14(2):163–181, 1999.

[27] A. Logg. Multi-adaptive error control for ODEs. Preprint 2000-003, Chalmers Finite Element Center, 2000.

[28] A. Logg. A multiadaptive ODE-solver. Preprint 2000-002, Chalmers Finite Element Center, 2000.

[29] G.I. Marchuk. *Adjoint Equations and Analysis of Complex Systems*. Kluwer Academic Publishers, Dordrecht, 1995.

[30] H.J.-P. Morand and R. Ohayon. *Fluid Structure Interaction*. John Wiley & Sons, Chichester, 1995.

[31] N.M. Newmark. A method of computation for structural dynamics. *ASCE Journal of Engineering Mechanics*, 85:67–94, 1959.

[32] N.A. Pierce and M.B. Giles. Adjoint recovery of superconvergent functionals from PDE approximations. *SIAM Review*, 42(2):247–264, 2000.

[33] B.M. Quadrelli and S.N. Atluri. Analysis of flexible multibody systems with spatial beams using mixed variational principles. *International Journal for Numerical Methods in Engineering*, 42:1071–1090, 1998.

[34] R. Rannacher. The dual-weighted-residual method for error control and mesh adaptation in finite element methods. In J.R. Whiteman, editor, *MAFELEAP 99*, pages 97–115, Amsterdam, 2000. Elsevier.

[35] R. Rannacher and C. Johnson. On error control in computational fluid mechanics. Technical report 94-13, Universität Heidelberg, 1994.

[36] A. Reister. *Ein Beitrag zur Berechnung des Schwingungsverhaltens gekoppelter Rotor-Fundament Systeme*. PhD thesis, Universität Kaiserslautern, 1998.

[37] R.D. Russell, D.M. Sloan, and M.R. Trummer. Some numerical aspects of computing inertial manifolds. *SIAM Journal of Scientific Computing*, 14(1):19–43, 1993.

[38] J.C. Simo and L. Vu-Quoc. On the dynamics in space of rods undergoing large motions: a geometrically exact approach. *Computer Methods in Applied Mechanics and Engineering*, 66:125–161, 1988.

[39] E. Süli and P. Houston. Finite element methods for hyperbolic problems: A posteriori error analysis and adaptivity. In I.S. Duff and G.A. Watson, editors, *State of the Art in Numerical Analysis*, Oxford, 1996. Oxford University Press.

[40] D.A. Spera. *Wind Turbine Technology*. ASME Press, New York, NY, 1994.

[41] R. Temam. *Infinite Dimensional Dynamical Systems in Mechanics and Physics*. Springer Verlag, Berlin, 1997.

[42] E.S. Titi. On approximate inertial manifolds to the Navier-Stokes equations. *Journal of mathematical analysis and applications*, 149:540–557, 1990.

Technische Universität Braunschweig
Informatik-Berichte ab Nr. 1998-04

1998-04	C. Lindig	Analyse von Softwarevarianten
1998-05	Gregor Snelting, Frank Tip	Reengineering Class Hierarchies Using Concept Analysis
1998-06	Juliana Küster Filipe	On a Distributed Temporal Logic for Modular Object Systems
1998-07	J. Schönwälder, M. Bolz, S. Mertens, J. Quittek, A. Kind, J. Nicklisch	SMX - Script MIB Extensibility Protocol Version 1.0
1998-08	C. Heimann, S. Lauterbach, T. Förster	Entwurf und Implementierung eines verteilten Ansatzes zur Lösung langrechnender Optimierungsprobleme aus dem Bereich der Ingenieurwissenschaften
1999-01	A. Zeller	Yesterday, my program worked. Today, it does not. Why?
1999-02	P. Niebert	A Temporal Logic for the Specification and Verification of Distributed Behaviour
1999-03	S. Eckstein, K. Neumann	Konzeptioneller Entwurf mit der Unified Modeling Language
1999-04	T. Gehrke, A. Rensink	A Mobile Calculus with Data
2000-01	T. Kaiser, B. Fischer, W. Struckmann	The Modula-2 Proving System MOPS
2000-02	J. Saperia, J. Schönwälder	Policy-Based Enhancements to the SNMP Framework
2000-03	A. Casties	Finite-Element-Interpolation der rumlichen Dichten eines Vielteilchensystems auf ungeordneten Gittern
2000-04	J. Koslowski	A 2-dimensional view of the Chu-construction
2000-05	S. Eckstein, P. Ahlbrecht, K. Neumann	Von parametrisierten Spezifikationen zu generierten Informationssystemen: ein Anwendungsbeispiel
2000-06	F. Strauß, J. Schönwälder, M. Mertens	JAX - A Java AgentX Sub-Agent Toolkit
2000-07	F. Strauß	Advantages and Disadvantages of the Script MIB Infrastructure
2000-08	T. Gehrke, U. Goltz	High-Level Sequence Charts with Data Manipulation
2000-09	T. Firley	Regular languages as states for an abstract automaton
2001-01	K. Diethers	Tool-Based Analysis of Timed Sequence Diagrams
2002-01	R. van Glabbeek, U. Goltz	Well-behaved Flow Event Structures for Parallel Composition and Action Refinement
2002-02	J. Weimar	Translations of Cellular Automata for Efficient Simulation
2002-03	H. G. Matthies, M. Meyer	Nonlinear Galerkin Methods for the Model Reduction of Nonlinear Dynamical Systems
2002-04	H. G. Matthies, J. Steindorf	Partitioned Strong Coupling Algorithms for Fluid-Structure-Interaction
2002-05	H. G. Matthies, J. Steindorf	Partitioned but Strongly Coupled Iteration Schemes for Nonlinear Fluid-Structure Interaction
2002-06	H. G. Matthies, J. Steindorf	Strong Coupling Methods
2002-07	H. Firley, U. Goltz	Property Preserving Abstraction for Software Verification
2003-01	M. Mayer, H. G. Matthies	Efficient Model Reduction in Non-linear Dynamics Using the Karhunen-Loève Expansion and Dual-Weighted-Residual Methods